KING SAUD UNIVERSITY
Information Systems Department.

IS230 Tutorial – Project Part II -Report Template
JDBC

Student ID(Leader):445102162
Student Name: نواف محمد الجليل

Student ID:445101358
Student Name: ريان محمد المشعل

Student ID:444102350
Student Name: سعد ناصر الدوسري

Student ID:445102004
Student Name: محمد عبدالله العجلان

Student ID:444101749
Student Name:محمد ال رشود

تنبيهات مهمة لا يعذر الطالب بعدم أخذها بعين الاعتبار مهما كان السبب:

1. **أي تشابه في المشاريع, و لو جزئي, يعرض الطالب لرصد صفر كدرجة للمشروع و تقرير لدكتور المادة مع إمكانية رفع التقرير لرئيس القسم.**
2. **عدم التقيد بمتطلبات التقرير (المذكورة في قالب تقرير المشروع (Project Report Template)) سوف يعرض الطالب لخصم درجات.**

1): Code for record **INSERTION**

```
while (true) {
    System.out.println("\nInserting a new employee:");

    System.out.print("EmployeeID: ");
    int id = scanner.nextInt();
    scanner.nextLine();

    System.out.print("Employee name: ");
    String name = scanner.nextLine();

    System.out.print("Salary: ");
    int salary = scanner.nextInt();

    System.out.print("Sales: ");
    int sales = scanner.nextInt();
    scanner.nextLine();

    String insertSQL = "INSERT INTO EMPLOYEES VALUES(" + id + ", '" + name + "', "
+ salary + ", " + sales + ")";
    stmt.executeUpdate(insertSQL);

    System.out.print("Insert another record (Y/N)?: ");
    String again = scanner.nextLine();
    if (!again.equalsIgnoreCase("Y")) break;
}
```

2): Code for **DISPLAYING** records

```
ResultSet rs = stmt.executeQuery("SELECT * FROM EMPLOYEES");

System.out.println("\nAll Employees:");
System.out.printf("%-12s | %-20s | %-10s | %-10s%n", "EmployeeID", "Name", "Salary",
"Sales");

while (rs.next()) { // <-- SQL interaction
    int id = rs.getInt("EmployeeID");
    String name = rs.getString("Name");
    int salary = rs.getInt("Salary");
    int sales = rs.getInt("Sales");

    System.out.printf("%-12d | %-20s | %-10d | %-10d%n", id, name, salary, sales);
}
```

3): Code for **INCREASING** the salary (Give yearly raises)

```
System.out.print("Enter sales goal: ");
int goal = scanner.nextInt();
scanner.nextLine();

ResultSet rs = stmt.executeQuery("SELECT * FROM EMPLOYEES");

System.out.println("\nYearly Raises:");
System.out.printf("%-12s | %-20s | %-12s | %-12s%n", "EmployeeID", "Name", "Old Salary", "New Salary");

while (rs.next()) {
    int id = rs.getInt("EmployeeID");
    String name = rs.getString("Name");
    int oldSalary = rs.getInt("Salary");
    int sales = rs.getInt("Sales");

    int newSalary = (sales >= goal) ? (int)(oldSalary * 1.10) : (int)(oldSalary * 1.05);

    String updateSQL = "UPDATE EMPLOYEES SET Salary = " + newSalary + " WHERE EmployeeID = " + id;
    stmt.executeUpdate(updateSQL);

    System.out.printf("%-12d | %-20s | %-12d | %-12d%n", id, name, oldSalary, newSalary);
}
```

3): Screenshots of the **EXECUTION**. Show the menu and examples for the 3 functions.

```
Choose an option:
1) Insert a new employee
2) Display all the employees
3) Give yearly raises
4) Exit
Choose an operation: 1

Inserting a new employee:
EmployeeID: 111
Employee name: Ahmed
Salary: 10000
Sales: 100000
```

```
Employees [Java Application] C:\Users\Admin\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_22.0.2.v20240802

Choose an option:
1) Insert a new employee
2) Display all the employees
3) Give yearly raises
4) Exit
Choose an operation: 2

All Employees:
EmployeeID   | Name               | Salary     | Sales
111          | Ahmed              | 10000      | 100000

Choose an option:
1) Insert a new employee
2) Display all the employees
3) Give yearly raises
4) Exit
Choose an operation:
```

```
Employees [Java Application] C:\Users\Admin\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_22.0.2.v2024080

Choose an option:
1) Insert a new employee
2) Display all the employees
3) Give yearly raises
4) Exit
Choose an operation: 3
Enter sales goal: 200000

Yearly Raises:
EmployeeID   | Name               | Old Salary   | New Salary
111          | Ahmed              | 10000        | 10500

Choose an option:
1) Insert a new employee
2) Display all the employees
3) Give yearly raises
4) Exit
```

4): All the code (الكود كاملا)

```
import java.sql.*;
import java.util.Scanner;

public class Employees {

   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);

      String url = "jdbc:mariadb://localhost:3306/Project2";
      String user = "root";
      String pwd = "";

      try {
         Connection con = DriverManager.getConnection(url, user, pwd);
  Statement stmt = con.createStatement();

         while (true) {
            System.out.println("\nChoose an option:");
            System.out.println("1) Insert a new employee");
            System.out.println("2) Display all the employees");
            System.out.println("3) Give yearly raises");
            System.out.println("4) Exit");
            System.out.print("Choose an operation: ");
            int choice = scanner.nextInt();
            scanner.nextLine();

            if (choice == 1) {

               while (true) {
                  System.out.println("\nInserting a new employee:");

                  System.out.print("EmployeeID: ");
                  int id = scanner.nextInt();
                  scanner.nextLine();

                  System.out.print("Employee name: ");
                  String name = scanner.nextLine();

                  System.out.print("Salary: ");
                  int salary = scanner.nextInt();

                  System.out.print("Sales: ");
                  int sales = scanner.nextInt();
                  scanner.nextLine();

                  String insertSQL = "INSERT INTO EMPLOYEES VALUES(" + id + ", '" +
name + "', " + salary + ", " + sales + ")";
                  stmt.executeUpdate(insertSQL);

                  System.out.print("Insert another record (Y/N)?: ");
                  String again = scanner.nextLine();
```

```
                if (!again.equalsIgnoreCase("Y")) break;
            }

        } else if (choice == 2) {
            ResultSet rs = stmt.executeQuery("SELECT * FROM EMPLOYEES");

            System.out.println("\nAll Employees:");
            System.out.println("EmployeeID  | Name          | Salary | Sales");

            while (rs.next()) {
                int id = rs.getInt("EmployeeID");
                String name = rs.getString("Name");
                double salary = rs.getDouble("Salary");
                double sales = rs.getDouble("Sales");
                System.out.println(id + " | " + name + " | " + salary + " | " + sales);
            }

        } else if (choice == 3) {
            System.out.print("Enter sales goal: ");
            double goal = scanner.nextDouble();
            scanner.nextLine();

            ResultSet rs = stmt.executeQuery("SELECT * FROM EMPLOYEES");

            System.out.println("\nYearly Raises:");
            System.out.println("EmployeeID  | Name  | Old Salary  | New Salary");

            while (rs.next()) {
                int id = rs.getInt("EmployeeID");
                String name = rs.getString("Name");
                double oldSalary = rs.getDouble("Salary");
                double sales = rs.getDouble("Sales");

                double newSalary = (sales >= goal) ? oldSalary * 1.10 : oldSalary * 1.05;

                String updateSQL = "UPDATE EMPLOYEES SET Salary = " + newSalary
+ " WHERE EmployeeID = " + id;
                stmt.executeUpdate(updateSQL);

                System.out.println(id + " | " + name + " | " + oldSalary + " | " + newSalary);
            }

        } else if (choice == 4) {
            System.out.println("Exiting program...");
            break;
        } else {
            System.out.println("Invalid option. Try again.");
        }
    }

    stmt.close();
    con.close();
```

```
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```